| INSERT Employee | INSERT INTO Employee (last\_name, first\_name, patronymic, role, salary, birth\_date, employment\_date, phone\_number, city, street, zip\_code)  VALUES ('${last\_name}', '${first\_name}', '${patronymic}', '${role}', ${salary}, '${birth\_date}', '${employment\_date}', '${phone\_number}', '${city}', '${street}', '${zip\_code}'); | Додати нового співробітника  CREATE TYPE ROLE AS ENUM ('manager', 'cashier'); | Manager |
| --- | --- | --- | --- |
| UPDATE Employee | UPDATE Employee SET last\_name = '${last\_name}', first\_name = '${first\_name}', patronymic = '${patronymic}', role = '${role}', salary = ${salary}, birth\_date = '${birth\_date}', employment\_date = '${employment\_date}', phone\_number = '${phone\_number}', city = '${city}', street = '${street}', zip\_code = '${zip\_code}' WHERE id = ${id}; | Оновити дані про співробітника за його id | Manager |
| DELETE Employee | DELETE FROM Employee WHERE id = ${id}; |  | Права видаляти надається тільки менеджерові |
| INSERT Category | INSERT INTO Category (name) VALUES ('${name}'); |  | Manager |
| UPDATE Category | UPDATE Category SET name = '${name}' WHERE id = ${id}; |  | Manager |
| DELETE Category | DELETE FROM Category WHERE id = ${id}; |  | Права видаляти надається тільки менеджерові |
| INSERT Product | INSERT INTO Product (name, description, category\_id) VALUES ('${name}', '${description}', ${category\_id}); |  | Manager |
| UPDATE Product | UPDATE Product SET name = '${name}', description = '${description}' WHERE id = ${id}; |  | Manager |
| DELETE Product | DELETE FROM Product WHERE id = ${id}; |  | Права видаляти надається тільки менеджерові |
| INSERT Store\_Product | INSERT INTO Store\_Product (upc, upc\_prom, selling\_price, amount, is\_promotional, product\_id)  VALUES ('${upc}', '${upc\_prom}', ${selling\_price}, ${amount}, ${is\_promotional}, ${product\_id}); |  | Manager |
| UPDATE Store\_Product | UPDATE Store\_Product SET upc = '${upc}', upc\_prom = '${upc\_prom}', selling\_price = ${selling\_price}, amount = ${amount}, is\_promotional = ${is\_promotional}, product\_id = ${product\_id} WHERE id = ${id}; |  | Manager |
| DELETE Store\_Product | DELETE FROM Store\_Product WHERE id = ${id}; |  | Права видаляти надається тільки менеджерові |
| INSERT Sale | INSERT INTO Sale (store\_product\_upc, amount, selling\_price, invoice\_id)  VALUES ('${store\_product\_upc}', ${amount}, ${selling\_price}, '${invoice\_id}'); | Продаж типу товару. Має атрибути UPC самого товару, кількості, загальної ціни та чеку, до якого належить | Cashier |
| DELETE Sale |  |  |  |
| INSERT Invoice | INSERT INTO Invoice (id, employee\_id, card\_id, total)  VALUES ('${id}', ${employee\_id}, '${card\_id}', ${total}); | Чек | Cashier |
| DELETE Invoice | DELETE FROM Sale  WHERE invoice\_id = '${id}';  DELETE FROM Invoice  WHERE id = '${id}'; |  | Права видаляти надається тільки менеджерові |
| INSERT Customer\_Card | INSERT INTO Customer\_Card (id, customer\_last\_name, customer\_first\_name, customer\_patronymic, phone\_number, city, street, zip\_code, percent)  VALUES ('${id}', '${customer\_last\_name}', '${customer\_first\_name}', '${customer\_patronymic}', '${phone\_number}', '${city}', '${street}', '${zip\_code}', ${percent}); | Карта клієнта | Manager |
| UPDATE Customer\_Card | UPDATE Employee SET last\_name = '${last\_name}', first\_name = '${first\_name}', patronymic = '${patronymic}', role = '${role}', salary = ${salary}, birth\_date = '${birth\_date}', employment\_date = '${employment\_date}', phone\_number = '${phone\_number}', city = '${city}', street = '${street}', zip\_code = '${zip\_code}' WHERE id = ${id}; |  | Manager/Cashier |
| DELETE Customer\_Card | DELETE FROM Customer\_Card WHERE id = '${id}'; |  | Права видаляти надається тільки менеджерові |
| —------------------------ | —------------------------ | —------------------------ | —------------------------ |
| **Касир** | | | |
| 4. Здійснити пошук товарів за назвою | SELECT \*  FROM Store\_Product sp  INNER JOIN Product p ON sp.product\_id = p.id  WHERE p.name = 'Apple'; | Searches for store\_product that references a product. |  |
| 5. Здійснити пошук товарів, що належать певній категорії, відсортованих за назвою | SELECT sp.upc, sp.selling\_price, sp.is\_promotional, p.name  FROM store\_product AS sp  JOIN product AS p ON sp.product\_id = p.id  JOIN category AS c ON p.category\_id = c.id  WHERE p.name = 'Apple' AND c.name = 'Fruits'  ORDER BY p.name DESC; |  |  |
| 6. Здійснити пошук постійних клієнтів за прізвищем | SELECT \* FROM Customer\_Card  WHERE customer\_last\_name = ${customer\_last\_name} |  |  |
| 9. Переглянути список усіх чеків, що створив касир за цей день | SELECT \*  FROM Invoice  WHERE employee\_id = ${employeeId} AND print\_date = CURRENT\_DATE; |  |  |
| 10. Переглянути список усіх чеків, що створив касир за певний період часу | SELECT \*  FROM Invoice  WHERE employee\_id = ${employeeId}  AND print\_date >= ${start\_date}  AND print\_date <= ${end\_date}; |  |  |
| 11. За номером чеку вивести всю інформацію про даний чек, в тому числі інформацію про назву, к-сть та ціну товарів, придбаних в даному чеку | SELECT \* FROM Sale  WHERE invoice\_id = ${invoice\_id}; |  |  |
| 12. Отримати інформацію про усі акційні товари, відсортовані за кількістю одиниць товару | SELECT \*  FROM Store\_Product  WHERE is\_promotional = true  ORDER BY amount DESC; |  |  |
| 12.1 Отримати інформацію про усі акційні товари, відсортовані за назвою | SELECT \*  FROM Store\_Product  WHERE is\_promotional = true  ORDER BY name DESC; |  |  |
| 13. Отримати інформацію про усі не акційні товарів, відсортовані за кількістю одиниць товару | SELECT \*  FROM Store\_Product  WHERE is\_promotional = false  ORDER BY amount DESC; |  |  |
| 13.1 Отримати інформацію про усі не акційні товари, відсортовані за назвою | SELECT \*  FROM Store\_Product  WHERE is\_promotional = false  ORDER BY name DESC; |  |  |
| 14. За UPC-товару знайти ціну продажу товару, кількість наявних одиниць товару. | SELECT selling\_price, amount  FROM Store\_Product  WHERE upc = ${upc}; |  |  |
| 15. Можливість отримати усю інформацію про себе. | SELECT \*  FROM EMPLOYEE  WHERE id = ${employee\_id}; |  |  |
| **Менеджер** | | | |
| 4. Видруковувати звіти з інформацією про усіх працівників, постійних клієнтів,  категорії товарів, товари, товари у магазині, чеки; | **Працівники** SELECT \*  FROM employee ORDER BY empl\_surname ASC; **Клієнти**  SELECT \*  FROM customer\_cards ORDER BY cust\_surname ASC; **Категорії товарів**  SELECT \*  FROM categories ORDER BY category\_name ASC;  **Товари**  SELECT \*  FROM products ORDER BY product\_name ASC; **Товари у магазині**  SELECT \*  FROM store\_products AS sp  JOIN product AS p ON sp.id\_product = p.id\_product ORDER BY product\_name ASC; **Чеки** SELECT \*  FROM invoices AS i  JOIN sales AS s ON i.invoice\_number = s.invoice\_number ORDER BY i.invoice\_number DESC; |  |  |
| 5. Отримати інформацію про всіх працівників, відсортованих за прізвищем | SELECT \*  FROM employee ORDER BY empl\_surname ASC; |  |  |
| 6. Отримати інформацію про всіх працівників, що займають посаду касира,  відсортованих за прізвищем | SELECT \*  FROM employee  WHERE empl\_role = 'cashier'  ORDER BY empl\_surname ASC; |  |  |
| 7. Отримати інформацію про всіх постійних клієнтів, відсортованих за прізвищем | SELECT \*  FROM customer\_cards  ORDER BY cust\_surname ASC; |  |  |
| 8. Отримати інформацію про всі категорії, відсортовані за назвою | SELECT \*  FROM category  ORDER BY category\_name ASC; |  |  |
| 9. Отримати інформацію про всі товари, відсортовані за назвою | SELECT \*  FROM product  ORDER BY product\_name ASC; |  |  |
| 10. Отримати інформацію про всі товари у магазині, відсортовані за кількістю | SELECT \*  FROM product AS p  JOIN store\_Product AS sp ON p.id\_product = sp.id\_product  ORDER BY sp.amount DESC; |  |  |
| 11. За прізвищем працівника знайти його телефон та адресу | SELECT phone\_number, city, street, zip\_code  FROM employee  WHERE empl\_surname = ${surname}; |  |  |
| 12. Отримати інформацію про усіх постійних клієнтів, що мають карту клієнта із  певним відсотком, посортованих за прізвищем | SELECT \*  FROM customer\_cards  WHERE percent = ${percent}  ORDER BY cust\_surname ASC; |  |  |
| 13. Здійснити пошук усіх товарів, що належать певній категорії, відсортованих за  назвою | SELECT sp.upc, sp.selling\_price, sp.is\_promotional, p.name  FROM store\_product AS sp  JOIN product AS p ON sp.product\_id = p.id  JOIN category AS c ON p.category\_id = c.id  WHERE p.name = 'Apple' AND c.name = 'Fruits'  ORDER BY p.name DESC; |  |  |
| 14. За UPC-товару знайти ціну продажу товару, кількість наявних одиниць товару,  назву та характеристики товару | SELECT sp.selling\_price, sp.amount, p.product\_name, p.characteristics  FROM Store\_Product sp  JOIN Product p ON sp.id\_product = p.id\_product  WHERE sp.UPC = ${upc}; |  |  |
| 15. Отримати інформацію про усі акційні товари, відсортовані **за кількістю одиниць**  **товару/ за назвою** | **За кількістю одиниць** SELECT \*  FROM Store\_Product  WHERE is\_promotional = true  ORDER BY amount DESC;  **За назвою** SELECT \*  FROM Store\_Product  WHERE is\_promotional = true  ORDER BY name DESC; |  |  |
| 16. Отримати інформацію про всі не акційні товари, відсортовані за кількістю  одиниць товару/ за назвою | **За кількістю одиниць**  SELECT \*  FROM Store\_Product  WHERE is\_promotional = false  ORDER BY amount DESC;  **За назвою**  SELECT \*  FROM Store\_Product  WHERE is\_promotional = false  ORDER BY name DESC; |  |  |
| 17. Отримати інформацію про усі чеки, створені певним касиром за певний період  часу (з можливістю перегляду куплених товарів у цьому чеку, їх назви, к-сті та ціни) | SELECT i.invoice\_number, print\_date, id\_employee, card\_number, print\_date, sum\_total, vat, s.upc, p.product\_name, s.product\_number, s.selling\_price  FROM invoice AS  JOIN sale AS s ON i.invoice\_number = s.invoice\_number  JOIN store\_product AS sp ON s.upc = sp.upc  JOIN product AS p ON sp.id\_product = p.id\_product  JOIN employee AS e ON i.id\_employee = e.id\_empolyee  WHERE e.id\_employee = ${idEmployee} AND e.empl\_role = ‘cashier’ AND print\_date >= :start AND print\_date < :end |  |  |
| 18. Отримати інформацію про усі чеки, створені усіма касирами за певний період  часу (з можливістю перегляду куплених товарів у цьому чеку, їх назва, к-сті та ціни) | SELECT i.invoice\_number, print\_date, id\_employee, card\_number, print\_date, sum\_total, vat, s.upc, p.product\_name, s.product\_number, s.selling\_price  FROM invoice AS i  JOIN sale AS s ON i.invoice\_number = s.invoice\_number  JOIN store\_product AS sp ON s.upc = sp.upc  JOIN product AS p ON sp.id\_product = p.id\_product  WHERE print\_date >= :start AND print\_date < :end |  |  |
| 19. Визначити загальну суму проданих товарів з чеків, створених певним касиром за  певний період часу | SELECT SUM(i.sum\_total) as revenue  FROM invoice AS i  JOIN employee AS e ON i.id\_employee = e.id\_employee  WHERE e.id\_employee = ${idEmployee} AND e.empl\_role = 'cashier' AND i.print\_date >= :start AND i.print\_date < :end |  |  |
| 20. Визначити загальну суму проданих товарів з чеків, створених усіма касирами за  певний період часу | SELECT SUM(sum\_total) as revenue  FROM сheck  WHERE print\_date >= :start AND print\_date < :end |  |  |
| 21. Визначити загальну кількість одиниць певного товару, проданого за певний  період часу | SELECT SUM(s.amount) as total\_units\_sold  FROM sale AS s  JOIN store\_product AS sp ON s.upc = sp.upc  JOIN product AS p ON sp.id\_product = p.id\_product  JOIN invoice AS c ON s.invoice\_number = i.invoice\_number  WHERE p.product\_name = ${productName} AND i.print\_date >= :start AND i.print\_date < :end |  |  |